

# Multiplexing Real Time Video Services

María Simon

Juan Pechiar

Instituto de Ingeniería Eléctrica, Facultad de Ingeniería, Montevideo, Uruguay

## Abstract

Statistical Bit Rate (SBR) ATM capability is considered a good option for supporting Variable Bit Rate (VBR) services. However, its study is somewhat in lag compared with Deterministic Bit Rate (DBR) or Available Bit Rate (ABR) capabilities. The definition of a general Call Acceptance Control (CAC) function is difficult to state for SBR.

We present some results regarding the multiplexing of real time data streams, mainly from interactive video services, which are naturally VBR and therefore candidates to use the SBR capability. It is shown that image quality is improved by using SBR instead of DBR. The coder design does not become more complicated. In fact, it remains the same.

We propose a statistical model for the traffic generated by such video sources. The delays introduced by a switch are studied following two approaches. Exact bounds are found for a worst case situation, indicating a very low statistical gain. However, simulations show that these bounds are too pessimistic since the worst case very rarely occurs. A very high mean load can be reached with acceptable delays. Indeed, the statistical gain is found to be significant. The CAC for this kind of service may be simple because, even assuming pessimistic figures, the burstiness for a real-time video data stream appears to be low.

*Keywords:* real time on ATM, packet video, SBR

## 1 Introduction

One major challenge for future ATM networks is the need to carry interactive services.

These have stringent real time requirements. Maximum end to end delays must be kept sufficiently small in order not to cause annoying effects (Section 4).

The amount of data video coders produce per time unit is not constant, but depends on instantaneous image complexity and activity. This amount can be controlled by varying the resulting image quality, thus allowing to fix a mean generation rate for example. Video coding and its control are described in Sections 2 and 3.

The produced data can be treated in several ways depending on the transmission scheme adopted and the service requirements.

In the case of a DBR connection, coded data can be buffered for transmission at constant bit rate (CBR). In general, the buffer size is quite important due to the variability of data generation. Such big buffers imply an important end to end delay so this method is more adequate for non interactive video, such as TV distribution.

In order to use smaller buffers, transmission at a variable rate (ideally the instantaneous data generation rate) would be desirable. This can be done on a DBR connection with the maximum bandwidth needed, but the channel will be misused. No statistical gain is obtained in the network.

The SBR capability (Section 5) guarantees a minimum data rate and also allows the emission at higher rates during limited periods. This makes SBR especially interesting for the transmission of interactive video services. Nevertheless, some shaping on the emitted traffic is still needed in order to comply with the SBR negotiated parameters.

In this case, the AAL2 (ATM Adaptation Layer) should be adopted to provide the necessary end to end synchronisation.

The use of ABR is not applicable in this case because traffic is controlled based on a user-network dialogue which is too slow for supporting this kind of traffic. No real time constraints are guaranteed.

So, SBR seems quite promising for carrying interactive video. Two main questions arise. First, is it possible to develop a simple generation controller which will take full advantage of SBR? Second, which will be the performance of a network carrying this type of traffic?

In previous work [8], we have studied the coding process for real time video services over SBR connections. It was shown that SBR conforming traffic allows for a better image quality than CBR at a given mean rate. Moreover, it was found that the CBR buffer control strategy could be maintained for SBR. These results are summarised in Section 6.

In order to answer the second question, a source characterisation is necessary. Software coding was performed on real sequences in order to find a parametrical characterisation of the SBR controlled video source (Section 7). From the obtained figures for mean rate and burstiness, an analytical worst case study is done in Section 8 which gives a very low statistical gain. However, simulations in Section 9 show that even a very highly loaded multiplexer exceeds the allowed delays with very low probability.

It is concluded that real time video transmission over the SBR capability provides quality improvement and statistical gain, due to the fact that the burstiness of the video source is low.

## 2 The video coder...

In this section we briefly describe the basics of MPEG video compression (see [4, 5, 6]) in order to show how the generated data is related to the video signal.

A video sequence is a series of frames, each being represented (in the digital domain) as a matrix of pixels. The transmission of all this data requires an enormous bandwidth. Fortunately, such data presents a lot of redundancy. Subsequent frames are normally very similar to each other, so a frame can be very well predicted from past frames, or even interpolated from the past and the future frames. This *temporal redundancy* is therefore reduced by coding only what is not predicted. This prediction is further improved by sending information on local displacements between both images (*motion compensation*). Only when big image transients occur (e.g. a scene cut) prediction fails.

Within a frame, there is also what is called *spatial redundancy*: a pixel's colour is normally similar to that of its neighbours. Therefore, the image is highly compressed by using spectral transforms (i.e. the *Discrete Cosine Transform* (DCT)). The DCT coefficients are then quantised using a step  $qp$  (quantisation parameter). The lower  $qp$ , the higher the coding quality, and the lower the compression factor.

Each frame is logically divided into *slices* and *macroblocks*. A macroblock (MB) is a  $16 \times 16$  pixel region where motion compensation is calculated. Macroblocks are formed by  $8 \times 8$  pixel blocks used for the DCT. A row of MB's form a slice.

Temporal prediction can be used in 3 ways in MPEG. Intra ( $I$ ) frames use no temporal prediction.  $I$  frames are normally located periodically in the data stream for editing purposes and for limiting transmission error propagation. These frames use only spatial compression, so they produce a large amount of data. In the case of real-time services,  $I$  frames would require large transmission buffers for storage, which would result in an unacceptable delay. Predictive ( $P$ ) frames are based on preceding  $P$  or  $I$  frames. Bidirectional or interpolated ( $B$ ) frames depend both on past and future  $I$  or  $P$  frames, thus resulting in a higher compression rate. The usage of  $B$  frames requires an extra delay for coding-decoding equal to the number of consecutive  $B$  frames used. This becomes unacceptable in the case of real time services.

## 3 ...and its controller

If the parameter  $qp$  is made constant, the video coder is said to be *uncontrolled* and the decoded image quality will be very consistent. But, the data generation of such a source is very difficult to

model: even the mean rate cannot be estimated beforehand. The study of this type of sources is therefore of limited practical interest. No reasonable traffic contract can be maintained during the communication, and bandwidth renegotiations are nowadays not applicable for real time services.

So we will base our studies on controlled video sources. In the remaining of the Section we describe the controller we used, first for the case of CBR traffic. In Section 6 we explain the applicability of this same controller for SBR.

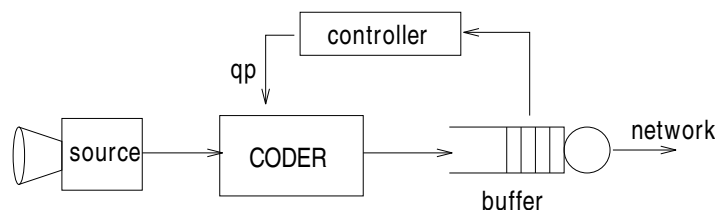


Figure 1: Controlled video coder

The aim of the controlled system is to make use of all the available bandwidth. For the CBR case this means transmission at a constant rate. Since data generation is variable, this is only possible by storing this data in a transmission buffer which absorbs the variations. If the mean generation rate equals the transmission rate, the buffer level will remain stable. This mean rate can be enforced by observing the transmission buffer level and acting on the quantisation parameter ( $qp$ ). When generation is excessive,  $qp$  can be made higher in order to reduce the generation rate, avoiding buffer overflow. The decoded image will be less accurate at this point. On the other hand, if the buffer is emptying,  $qp$  can be lowered.

The controller can act on a frame, slice or macroblock basis. Smaller control units result in more stable buffer levels, but less consistent image quality.

The mean output rate determines the overall quality, while the buffer size affects the response to transients as scene cuts or fast movements. If the buffer is large enough, such transients can be coded with sufficient quality levels. Small buffers require an instantaneous degradation which will take several frames for recovery.

## 4 The real time framework

Video services with real time constraints are those in which a low end to end delay must be satisfied. Examples of this type of services are videotelephony or videoconference.

This work refers to good quality interactive video services. This means about SIF resolution (Source Input Format,  $288 \times 352$  pixels), 25 or 30 non interlaced frames per second, continuous movement, and signal to noise ratio of at least 20 dB. The image quality is expected to be as consistent as possible.

In these interactive services acquisition to presentation delays must be kept below 150 ms to 200 ms in order not to confuse the users. With higher levels of delay, the fluidity of the conversation becomes affected.

When these services are carried over a data network, several factors contribute to total delay:

- acquisition time
- coding and decoding, at least one slice time, due to video scanning
- transmission and reception buffering, which is one problem focused in this work
- propagation, which implies low delays even in nation wide communications
- switching delay, caused by queuing in the multiplexers. This kind of delays is analysed in this work as well

Taking into account that the end to end delay must not exceed 150 ms, a delay of 80 to 100 ms is assigned for buffering purposes. This limits the transmission buffer size.

Switching delays are studied in sections 8 and 9. They are usually lower than buffering delays since cell rates throughout the network are much higher than the application's cell rate. Normally the cells will cross a set of switches. It is to be assumed that transit switches are handling virtual paths, that constitute almost DBR bundles. Therefore, the delay in transit switches will be less than in the access switches. We adopt a design value for each queuing delay to be less than ten milliseconds.

The coding and decoding delay is low since only predictive ( $P$ ) frames are coded.  $I$  and  $B$  frames are avoided because of the reasons exposed in Section 2. To ensure the error concealment without a high generation peak, the coder used in this work implements a gradual refresh procedure. A sliding column of macroblocks is Intra coded every certain number of frames. A column instead of a slice was preferred because the extra generation is more uniformly distributed. Motion compensation is forbidden across the sliding column.

## 5 The SBR capability

The SBR capability is defined by the ITU-T in [7], as intended to support VBR sources and allow for statistical gain. The traffic is described by means of the Sustainable Cell Rate (SCR), the Peak Cell Rate (PCR) and Maximum Burst Size (MBS).

PCR is the maximum rate at which cells can be emitted by the source. SCR is the maximum mean cell rate and MBS is the maximum size (in cells) a burst emitted at PCR can have.

Cell conformance is tested at the Usage Parameter Control (UPC) by a Generic Cell Rate Algorithm (GCRA). The most popular implementation of the CGRA is the leaky bucket which is a fluid equivalent algorithm.

Non conforming cells can be discarded by the network. This is not acceptable for video transmission because random losses in the data stream cause important degradation on the decoded image. It is therefore preferable to reduce image quality at the coder side ("graceful degradation") and send conforming cells than to risk random cell losses in the network.

Call Acceptance Control procedures are difficult to define for an SBR environment. The network must guarantee a certain quality of service (QoS) while obtaining some statistical gain. Meeting these goals simultaneously is not obvious for such poorly characterised sources.

## 6 Adapting the video coder to SBR connections

Transmission of interactive video over an SBR connection seems quite promising in several aspects. Burst emission can be used to avoid transmission buffer overflow when a scene transient occurs, therefore allowing smaller buffers and therefore less delay, or more consistent quality because data can be generated as if a larger buffer existed.

We will show that this is actually true. How the complexity of the coding system is affected is treated below, where we summarise the main results from [8]. There are still synchronisation problems to be solved by the AAL 2, which are not addressed in this work.

### 6.1 Adapting the controller

At first glance, as we established that only conformant cells may be emitted, the design a controller for SBR transmission seems a difficult issue. We wish to take full advantage of the SBR capability and, simultaneously, ensure that the generated data will always reach the receiver before its presentation time. As the allowed transmission depends on the available credits, it varies during the communication. An image of the conformance testing algorithm seems to be needed.

In addition, given a data amount to be transmitted and a number of available credits, there is still a choice on the actual usage of the credits and on the storage in the transmission buffer. For example, the user can transmit as much as it can at PCR, and continue transmission at SCR when credits are exhausted. Or it can transmit at SCR and use credits only if the generated amount exceeds the transmission buffer. We refer to these delivery rules as *transmission policies*.

In our previous work ([8]) we studied the controller for an SBR environment, and we presented a control strategy that avoids the need of knowing the transaction status, and is independent of the actual transmission policy used. It is found that if a couple of bounds are satisfied, the emitted cells will be conformant.

The following notation will be used:

**SCR** sustainable cell rate

**PCR** peak cell rate

**MBS** maximum burst size

**B** bucket size

**TB** transmission buffer level

**Delay** the maximum buffering delay

**C** used credits.  $0 < C < B$

These parameters are not independent. For the fluid approach they are related by:  $B = (1 - SCR/PCR)MBS$ .

During the communication the transmission buffer level must satisfy the bounds:

$$TB < Delay \times PCR \quad (1)$$

$$TB < Delay \times SCR + B - C \quad (2)$$

If (1) does not hold, data may not arrive on time to the receiver even when transmitted at maximum rate.

If (2) does not hold, there are not enough credits to transmit the data volume.

With normal parameter values (2) is more stringent than (1). This will always happen if  $B < Delay(PCR - SCR)$ , or

$$MBS < Delay \times PCR \quad (3)$$

When MBS is kept below this bound, only (2) must be verified during the coding process, which is equivalent to

$$TB + C < Delay \times SCR + B \quad (4)$$

Condition (4) is ensured by controlling a virtual buffer level  $TB + C$ . Just as in the case of CBR, this value grows with generation and decreases at constant rate SCR. Its capacity, which is the constant given in the right hand side of (4), shows that the bucket behaves as an additional storage (recall that for CBR, the buffer had a maximum  $Delay \times mean\ rate$ ). This means that the controller used for CBR can now be used for SBR by observing a virtual buffer level rather than the actual transmission buffer level. And it is very simple to keep track of this value.

The conditions (1) and (3), typical of a real time service, state a limit on the traffic burstiness. Indeed, the generation data rate is limited because the signal acquisition is also in real time. Therefore, the user will not demand a very high PCR because it would be useless. As the delay is also limited, the maximum burst is relatively low.

SCR determines the steady state quality. The virtual buffer capacity determines the transient response.

## 6.2 Output traffic

We found in different situations the minimum mean rate that allows the communication, for a given total buffering delay (80 ms). The difference is significant. A CBR connection can honour the required delay with a rate of over 380 kbps. If the emission of bursts is allowed and controlled by a leaky bucket algorithm, the minimum operation rates are as shown in Table 1. These figures were obtained from a particular sequence and coder, and we do not present them as a general

connection type	SCR(kbps)	PCR(kbps)	MBS(kbits)
CBR	392	392	
SBR	124	2000	64
SBR	107	2000	160

Table 1: Minimum operation rates to achieve transmission with an 80 ms delay

result. However, the difference between CBR and SBR allows to conclude that the minimum mean rates will be in general lower for SBR.

It is to be noted, in Table 1, that enlarging the bucket size result in a slight increase in the minimum operation rate. The burst size of 160 kbits is the maximum compatible with the delay, for a PCR = 2 Mbps, according to restriction (1).

The quality of the sequences from Table 1 is obviously poor, because those rates are absolute minimums. The quality comparison was performed by coding a given sequence under both CBR and SBR environments with equal mean rate. This mean was 400 kbps, higher than the minimum.

### 6.3 Quality results

We worked on real sequences, including moderate movement and scene cuts. Data presented in this article correspond to a sequence formed by “Claire” followed by “Foreman”.

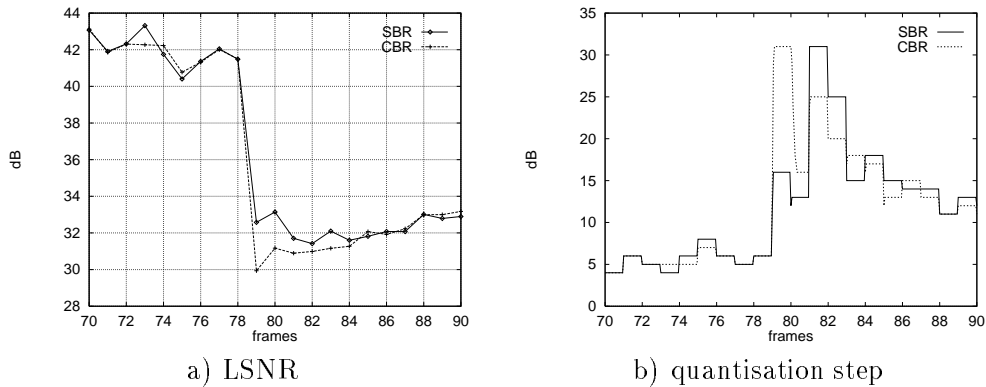


Figure 2: Quality comparison between CBR and SBR

The quality consistency was improved by operating in an SBR environment, as shown in Figure 2a, which depicts the Luminance Signal to Noise Ratio (LSNR) during a scene cut. Both mean rates are equal. Perceptual evaluation reveals a significant difference. Even though the overall generation volumes should be maintained, the possibility of emitting a burst allows to build a fairly good new image when the new sequence begins. Subsequent frames, in which generation is reduced to enforce the mean rate, are then very well predicted. The  $qp$  values are shown for CBR and SBR in Figure 2b. In SBR, high  $qp$  values are used some frames later. Figure 3 shows the error images (difference between decoded and original images) for the first frame after the scene cut. It can be seen that the emission of relatively low bursts is reflected in an improvement in the quality consistency.

So, for a same mean rate, SBR allows better response to transients. On the other hand, given a delay requirement, SBR will allow much lower mean rates than CBR (at the expense of a lower mean quality). This will require less network resources, and increase network performance in terms of number of connections.

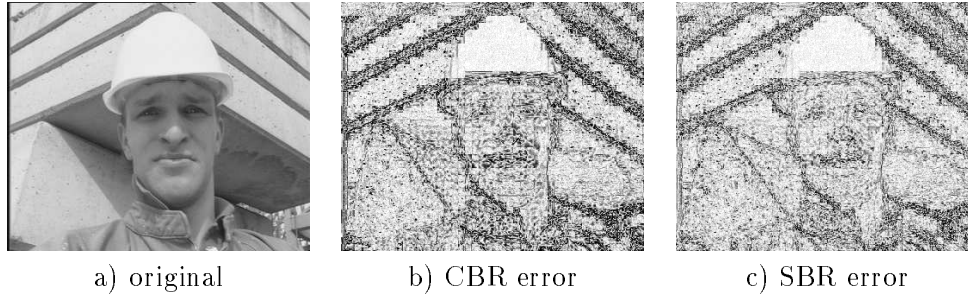


Figure 3: Error images

## 7 Source characterisation

Even when a video source is controlled as in our case, the traffic it generates has a very complex shape. We adopt simplified models and choose rather pessimistic parameters in order to obtain conservative bounds.

Normally a source will be emitting around SCR until an important transient occurs. In this moment, a large amount of data will be generated. This means that the virtual buffer level will grow near its maximum and a high rate burst has to be emitted (otherwise this data will not arrive at the decoder for presentation time). The volume of this burst approximately corresponds to an Intra coded frame. In our case we got values of about 40 kbit.

After emitting a burst, the controller will reduce the mean generation rate in order to recover the wasted credits (or equivalently, the virtual buffer's target level). This is a main characteristic of SBR: this period of recovery is necessary to force the mean rate to no more than SCR.

In the case of interactive video, these bursts are generally caused by for example: camera switching, the speaker showing a document to the camera or an image from another source, sudden and fast movements. The mean time between such events was taken 10 s. This is a very pessimistic figure because such events are much more rare.

Statistical independence between communications is to be expected, due to the nature of conversational services.

So, we propose 2 models:

**4 state model:** a source emits normally at around SCR. Every 10 s in average a burst is triggered. The volume of the burst is distributed around 40 kbits. This burst is evacuated at PCR until no more credits are left, so emission continues at SCR. Then there is a recovery period (with mean rate 0.8 SCR) after which SCR transmission resumes. In this model, the mean rate is forced to SCR and burst sizes are limited, much like in an SBR connection.

**2 state model:** this simpler model was used because it is more analytically tractable. A source emission is modelled as a 2 state Markov Modulated Poisson Process (MMPP). In the active state, the mean emission rate is 960 kbps, and mean duration 1/25 s. The normal state has mean duration 10 s and a mean rate which is calculated so that the global mean rate of the MMPP is SCR. Note that in this case, even if the mean rate is SCR, this rate is not forced in the short term as in the other model. This means that for a highly loaded multiplexer, this type of source will surely result in a much higher queue congestion.

## 8 Worst case analysis

Given the connection parameters SCR, MBS and PCR, deterministic bounds can be found for the maximum delay in a network element when several similar data streams are multiplexed. This deterministic approach is based on a worst case hypothesis, and is presented as a fluid approximation since cells are small compared to data volumes generated.

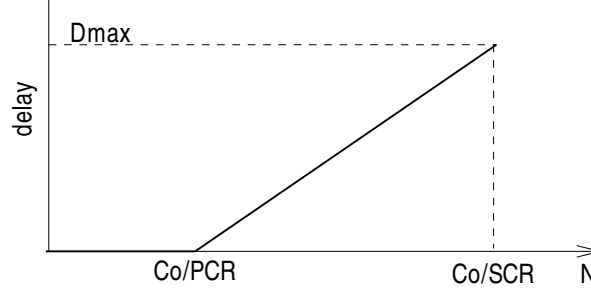


Figure 4: Multiplexer delay vs. number of sources

We used the general results of Cruz [1, 2] on burst constrained traffic to derive a bound on delay in a FCFS multiplexer.

We will consider a burstiness constraint that takes into account the parameters we have estimated: SCR, PCR, and MBS. This bounding function is very similar to the  $(\sigma, \rho)$  proposed by Cruz, with the additional restriction of PCR. The bucket size  $B$  plays the role of  $\sigma$ , and SCR corresponds to  $\rho$ . This constraint is satisfied by any traffic that has passed a leaky bucket shaper with parameters  $(B, SCR)$  and maximum output speed PCR. The bucket size is calculated as  $B = MBS(1 - \frac{SCR}{PCR})$ .

If we consider  $N$  streams with rate functions as described entering a FCFS multiplexer with an output link speed  $C_0$ , we can derive a bound on the maximum delay of any data bit going through the multiplexer.

If  $N \times PCR < C_0$ , then no backlog can build up except at cell scale. The multiplexer would be very lightly loaded. This is the same situation as if the traffic was sent using a DBR channel with rate PCR, normally misused, to guarantee a very low delay.

For the system to be stable,  $N < N_{max} = C_0/SCR$ . In the interesting case ( $N \times SCR < C_0 < N \times PCR$ ), the delay  $D$  is bounded by:  $D < \frac{N}{N_{max}} \frac{B}{SCR}$ . This bound is reached if all sources transmit simultaneously their maximum burst MBS at rate PCR. When  $N \sim N_{max}$ , the effective rate at the output link for each source is SCR, and the maximum delay is the same as if the application buffer absorbed the burst, transmitting therefore at CBR = SCR.

For the parameters MBS = 64000 bits, SCR = 400 kbps, PCR = 960 kbps, the maximum delay becomes 93 ms. This is clearly unacceptable (this is the delay only in the first multiplexer).

These results show that the analysis of multiplexing SBR video sources using a worst-case approach does not show significant advantages over DBR.

## 9 Simulation of a multiplexer

### 9.1 Four phase source model

The worst-case situations studied above happen very rarely. A good idea of which could be the delays introduced by multiplexing several video SBR connections was obtained through a set of simulations.

Parameters for the 4-state model described above were as follows: The burst size is normally distributed with a mean of 40 kbit, and a standard deviation of 5 kbit. The time between scene cuts has a Cox distribution, in this case the sum of 3 exponentials. The mean has been adjusted to 10 s, which is a very short time, and therefore produces a more bursty traffic than a real source.

Several of these sources are fed into a FCFS queue which acts as a multiplexer with output capacity  $C_0$ . These sources have the parameters: SCR corresponding to 400 kbps, PCR corresponding to 960 kbps and MBS to 64 kbit.

As a result, we obtain an estimate for the probability that the backlog in this queue exceeds a certain delay.



Different loads were studied, reaching 97.5% average load. Figure 5a shows the probability of exceeding each delay. It is to be noted that for high loads the curves exhibit the two slopes behaviour described in [3], corresponding to the cell and burst congestion scales.

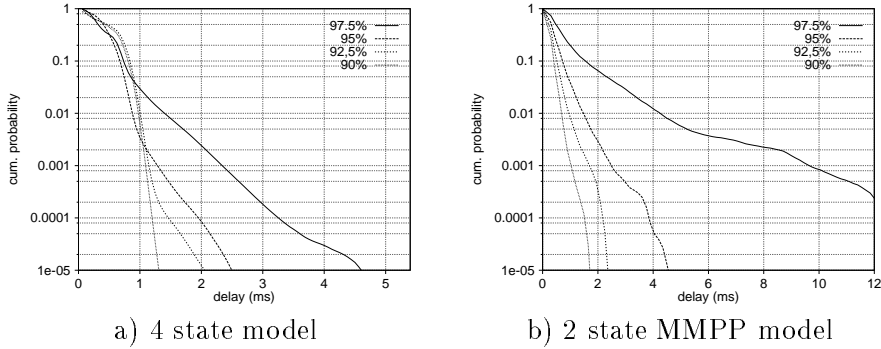


Figure 5: Simulation results  $C_0 = 16$  Mbps,  $SCR = 400$  kbps

We can see from the plots in Figure 5a that even with a load of 97.5%, delays are greater than 5 ms with probability less than  $10^{-5}$ .

## 9.2 Two phase source model

For this model, resulting delays are, as expected, substantially higher, especially for the highest load (Figure 5b). In [3] an analytical approximation is presented for the superposition of ON-OFF sources that is also valid if a Poissonian traffic is added. In our case this Poissonian traffic has a mean value equal to the mean of the MMPP normal state. The aggregation of these sources is approximated by a single MMPP with only 2 states which correspond to the instantaneous traffic entering the multiplexer being higher (overload state) or lower (underload state) than its output rate.

Solving this model in our case indicates that effectively, even with a highly loaded system, overload periods are relatively unimportant. For example, in the 97.5% case, overload periods occur 1/7 of the time, and with a cell arrival rate which is only 1% higher than  $C_0$ .

## 10 Conclusions

The SBR capability, compared with DBR, allows to obtain a better quality when other conditions (mean rate, delay) remain equal. Real time services can be implemented over SBR channels with moderate SCR, PCR and MBS values. A generation controller that takes advantage of the SBR transfer capability is easily implemented.

The multiplexing performance is high: even assuming pessimistic models and parameters, the delays remain within the design ranges with high probability.

The CAC function is not critical, because real time services cannot generate very important bursts, mainly because these applications cannot generate data at arbitrary high rate. The equivalent bandwidth is very close to the mean rate, giving therefore a simple estimation of the resources that can be required for each user.

Non interactive video has a periodic generation pattern caused by Intra frames. Those bursts are strongly correlated, and therefore our analysis is not valid. In these cases no real time constraints are to be met. Therefore a big smoothing buffer may be used, and transmission will use DBR connections, eventually with bandwidth renegotiation.

## Acknowledgements

This work was supported by projects from the CONICYT (National Council for the Research and Development) and the CSIC (University's Commission for Research Support).

## References

- [1] R.L. Cruz, A Calculus for Network Delay, Part I: Network Elements in isolation, *IEEE Trans. on Information Theory*, Jan. 1991, Vol. 37, Num. 1.
- [2] R.L. Cruz, A Calculus for Network Delay, Part II: Network Analysis, *IEEE Trans. on Information Theory*, Jan. 1991, Vol. 37, Num. 1.
- [3] A. Baiocchi, N. Blefari Melazzi, M. Listani, A. Roveri, R. Winkler. Loss performance analysis of an ATM multiplexer loaded with high speed ON OFF sources. *Journal of Selected Areas in Communications*, April 1991, pp. 388 393.
- [4] D. Le Gall. The MPEG Video Compression Algorithm: A Review. *SPIE Vol. 1452, Image Processing Algorithms and Techniques II*. 1991.
- [5] ISO/IEC JTC 1, International Standard 11172, (MPEG 1).
- [6] ISO/IEC 13818 Generic coding of moving pictures and associated audio. (MPEG 2). Singapore, Nov. 1994.
- [7] ITU-T Recommendation I.371, Traffic control and congestion control in B-ISDN. Geneva, Study Group 13 meeting, July 1995.
- [8] M. Simon, J. Pechiar, M. de Oliveira, L. Casamayou. Video coding and ATM statistical bit rate capability. *ATM Networks. Performance Modelling and Analysis*. Ed. D. Kouvatsos. Chapman & Hall. To be published.